Remarks/Arguments

With reference to the Office Action of April 7, 2006 Applicants offer the following remarks.

Non-Art Rejections and Objections

In the Office Action of April 7, 2006,

1. Claims 1-12 were rejected under 35 USC §101 as directed to non-statutory subject matter in that there is no recitation of any outcome result suggesting what the calculations are being applied to.

2. Claims 7-9 were rejected under 35 USC §101 as directed to non-statutory subject matter in that there is no recitation of a computer readable medium.

3. There were a plurality of rejections under 35 USC §112 (second paragraph) which have been dealt with by suitable amendment.

Art Rejections

All of the pending claims have been rejected as anticipated, 35 USC §102(e), by United States Patent 6,510,406 to Marchisio For Inverse Inference Engine For High Performance Web Search.

The gist of Marchisio, as applied, is the management of uncertainty (imprecision) in the user input. The input is a "user query vector" having as many elements as the number of rows in a "term spread matrix." These reduce the documents and queries to a symbolic form. The query vector is used to generate an error covariance matrix for dealing with the uncertainty. This is used to get to a covariance matrix of errors in the input query vector, thereby allowing for user input errors in the input query vector.

Marchisio also show "clustering" of lexically related documents, as well as the use of "positive bases" and "negative bases." "Negative bases" may not necessarily contain desired keywords, but may contain a statistically significant number of keywords conceptually related to the keywords in the user query.

Marchisio was applied to original Claim 1 as follows:

A data retrieval system for causing a computer to retrieve data being stored in a database, said retrieval system comprising:

a database storing data as a vector digitized based on a keyword;

> At step 16, the disclosed system receives a user query from a user, consisting of a list of keywords or phrases. The disclosed system parses the electronic text included in the received user query at step 16. The parsing of the electronic text performed at step 16 may include, for example, recognizing acronyms, extracting word roots, and looking up those previously generated concept ID numbers corresponding to individual terms in the query. In step 17, in response to the user query received in step 16, the disclosed system generates a user query vector having as many elements as the number of rows in the term-spread matrix generated at step 9. (emphasis added) (col 7, lines 27-38)

Further as illustrated in FIG. 2, the indexing module 20 performs steps to reduce the original documents 27 and a query received from one of the clients 21 into symbolic form (i.e. a term-document matrix and a query vector, respectively). (Emphasis added) (Col. 8, lines 64-67)

a means for generating a residual vector from said data to compute a covariance matrix and an eigenvector of said covariance matrix, and for generating and storing a set of basic vectors from a set of said computed eigenvectors;

Following creation of the query vector at step 17, at step 18 the disclosed system generates, in response to the user query vector, an error-covariance matrix. The error-covariance matrix generated at step 18 reflects an expected degree of uncertainty in the initial choice of terms by the user, and contained within the user query. (Emphasis added) (Col. 7, lines 39-45)

S.sub.ij is the covariance matrix of the errors n.sub.i in the input query vector, computed as S.sub.ij =Covar[n.sub.i,n.sub.j]=.delta..sub.ij n.sub.i.sup.2, if we assume that the errors n.sub.i on the elements of the input query are independent. By allowing for errors in the input query vector, which is based on the terms in the original query, the present system attaches a margin of uncertainty to the initial choice of terms input by the user. Since the user's initial term selection may not be optimal, the present system advantageously allows for a margin of error or a certain degree of flexibility in this regard. (Emphasis added) (Column 14, lines 26-34)

LSI transforms the matrix D as D'=U.sub.k.LAMBDA..sub.k V.sub.k.sup.T where .LAMBDA.=diag(.lambda..sub.1, . . . ,.lambda..sub.k) and {.lambda..sub.i,i=1,k} are the first k ordered singular values of D, and the columns of U.sub.k and V.sub.k are the first k orthonormal eigenvectors associated with DD.sup.T and D.sup.T D respectively. From this we see that .omega.=(U.LAMBDA.).sub.k and A=V.sub.k.sup.T {A.sub.j, j=1,2, . . . ,n}.(Col. 12, lines 1-7)

FIG. 5 shows clustering for the document collection reflected by the table of FIG. 4, as obtained using an LSI approach, as in some existing systems. The dots in each of the graphs in FIG. 5 are plane projections of individual documents into "concept space", as determined by a choice of the first few eigenvectors. Documents which deal with similar topics cluster together in this space. (Emphasis added) (Col. 11, lines 20-25)

a means for reading out said data and at least one of said eigenvectors from a memory, and for computing a contribution of said eigenvector to said data, and for contracting or enlarging a residual vector to store; and

FIG. 5 shows clustering for the document collection reflected by the table of FIG. 4, as obtained using an LSI approach, as in some existing systems. The dots in each of the graphs in FIG. 5 are plane projections of individual documents into

"concept space", as determined by a choice of the first few eigenvectors. Documents which deal with similar topics cluster together in this space. (Col. 11, lines 20-25)

where w=A.sup.T.alpha. is the smallest l.sub.2 norm solution to the linear system Dw=q. Reducing the number of eigenvectors in the approximation to the inverse of D has a regularizing effect on the solution vector w, since it reduces its norm. (Column 12, lines 15-18)

S.sub.ij is the covariance matrix of the errors n.sub.i in the input query vector, computed as S.sub.ij =Covar[n.sub.i,n.sub.j]=.delta..sub.ij n.sub.i.sup.2, if we assume that the errors n.sub.i on the elements of the input query are independent. By allowing for errors in the input query vector, which is based on the terms in the original query, the present system attaches a margin of uncertainty to the initial choice of terms input by the user. Since the user's initial term selection may not be optimal, the present system advantageously allows for a margin of error or a certain degree of flexibility in this regard. (Emphasis added) (Column 14, lines 26-34)

a means for selecting a keyword to be used for labeling according to a similarity between said stored basic vector and said data, and a weight on said similarity so as to store the keyword in a memory.

FIG. 4 shows an example of a term-document matrix 40, and also illustrates some of the difficulties associated with existing systems. The term-document matrix 40 of FIG. 4 is shown, for purposes of illustration, loaded with word counts for 16 keyword terms (rows 42) in 15 documents (columns 44). The example of FIG. 4 illustrates testing of latent semantic retrieval. Topics present in document collection of FIG. 4 are "GEOGRAPHY" (documents b3, b4, b6 and b12), "THEATER" (b1, b5, b8, b9, b10, and b15), and "SHAKESPEARE" (b7 and b11). The keyword "Shakespeare" appears only in documents b7 and b11. The documents semantically related to the "THEATER" topic, however, may also be relevant to a search query which includes the single keyword "Shakespeare".

FIG. 5 shows clustering for the document collection reflected by the table of FIG. 4, as obtained using an LSI approach, as in some existing systems. The dots in each of the graphs in FIG. 5 are plane projections of individual documents into "concept space", as determined by a choice of the first few eigenvectors. Documents which deal with similar topics cluster together in this space. The key to successful semantic retrieval is to select a subspace where documents 54 which contain the keyword "Shakespeare" cluster as a subset of all documents 56 which deal with the topic of "THEATER". This is the case for the two projections shown by the graphs 50 and 52, but not for graphs 51 and 53. Graphs 51 and 53 in FIG. 5 are examples where the "SHAKESPEARE" documents 54 do

not appear as a subcluster of the "THEATER" documents 56. Graphs 50 and 52, on the other hand, are examples where the "SHAKESPEARE" documents 54 appear as a subcluster of the "THEATER" documents 56. It is difficult to predetermine which choice of projection axes x-y that will cause the desired effect of clustering the "SHAKESPEARE" documents as a subcluster of the "THEATER" documents. More specifically, it is difficult to predetermine how many eigenvectors--and which ones--one should use with LSI in order to achieve this result. FIG. 5 illustrates that there is no way of pre-determining the combination of axes which cause the "SHAKESPEARE" documents to appear as a subcluster of the "THEATER" documents. (Emphasis added) (Column 11, lines 4-46)

The graph 62 of FIG. 6 illustrates the comparison of the desired output query q (solid line 63) and the computed output query q' (undistinguishable from q) for the l.sub.2 -norm minimizing solution. The output q' is computed as a linear superposition of the first seven bases (also shown in FIG. 6), ordered by decreasing coefficients .vertline..alpha..sub.i.vertline.. Bases with positive .alpha..sub.i (basis 1 and basis 2) are shown with continuous lines. Bases with negative .alpha..sub.i (basis 3, basis 4, basis 5, basis 6, and basis 8) are shown with dotted lines. The positive bases contain primarily the input query keyword and contribute significantly to the query approximation. They also contain several other keywords (e.g. "theatre", "comedy") which are directly associated with the keyword "Shakespeare" across the document collection. These associated keywords must be subtracted in order for the approximation q' to match the desired output q. The negative bases accomplish this. The negative bases define partitions (or groups) of documents that contain many of the same keyword patterns found in the positive bases, this time never in direct association with the keyword "Shakespeare". Consequently, the negative bases span the space of the latent semantic documents. Latent semantic documents are documents that, while not containing any of the keywords in the user query, may contain a statistically significant number of keywords conceptually related to the keywords in the user query. (Emphasis added) (Column 15, lines 25-50)

Specifically, at step 14, the original term-document matrix created at step 6 and potentially weighted at step 8, rather than the term spread matrix computed at step 9, is cross-multiplied with the unsorted document weights generated at step 12 (note that the document weights must be unsorted in this step to match the original order of columns in the term-document matrix) to form a plurality of term weights, one for each term. These term weights reflect the degree of correlation of the terms in the lexical knowledge base to the terms in the user query. (Column 8, lines 23-32)

Art of Record

The only reference of record is US Patent 6,510,406 to Marchisio For Inverse Inference Engine For High Performance Web Search. Marchisio describes an information retrieval system that deals with uncertainty, i.e., the problems of synonymy, polysemy, and retrieval. This is said to be accomplished by allowing for a wide margin of uncertainty in the initial choice of keywords in a query. That is, for each input query vector and an information matrix, Marchisio's system solves an optimization problem which maximizes the stability of a solution at a given level of misfit. The disclosed system may include a decomposition of the information matrix in terms of orthogonal basis functions. Each basis encodes groups of conceptually related keywords. The bases are arranged in order of decreasing statistical relevance to a query. The disclosed search engine approximates the input query with a weighted sum of the first few bases. Other commercial applications than the disclosed search engine can also be built on the disclosed techniques.

Discussion

Status of the Claims.

Claims 1-20 were originally presented for Examination. These claims were subject to a restriction requirement. Applicants elected claims 1-12. All of the elected claims were rejected in the Office Action of April 7, 2006. Applicants have amended their claims to particularly point out and describe their invention, and distinguish over the art of record.

Discussion: Non-Art Rejections

Applicants have amended their claims to distinctly point out and describe their invention, and to obviate the non art rejections.

Discussion: Art Rejection

The overarching issue is whether the claims, as limited by the newly added clauses and limitations are allowable over the art of record. Narrowly defined, the issue is whether art describing negative bases to define partitions teaches or suggests a contribution vector (defined in connection with an inner product calculated between a calculated vector and a document vector) to calculate a residual matrix to enhance document retrieval.

Exemplary Claim

Claim 1, as amended, is exemplary:

> *1 (Currently Amended) A data retrieval system for causing a computer to retrieve data being stored in a database, said retrieval system comprising: a database storing data as a vector digitized based on a keyword; a means for generating a residual vector from said data, said residual vector corresponding to a vector in which an element corresponding to a contribution component in a direction of a basic vector calculated is subtracted from a previously obtained residual vector, the basic vector and the residual vector a newly generated residual vector lie in an orthogonal relationship, to compute a covariance matrix and an eigenvector of said covariance matrix, and for generating and storing a set of basic vectors from a set of ~~said~~ computed eigenvectors; a means for reading out said data and at least one of said eigenvectors from a ~~memory~~ database, and for computing a contribution vector of said eigenvector to said data, and for contracting or enlarging a residual vector to store; ~~and~~ a means for selecting a keyword to be used for labeling clusters according to a similarity between said stored basic vector and said data, and a weight on said similarity so as to store the keyword in a ~~memory~~ database; means for classifying data into clusters of documents having the same or similar keywords and depending on a similarity between the stored basic vector and the data; and a means for outputting cluster data of a cluster to a graphical user interface system for displaying the cluster data.*

A data retrieval system for causing a computer to retrieve data being stored in a database, said retrieval system comprising:

a database storing data as a vector digitized based on a keyword;

which has been amended thusly:

1 (Currently Amended) A data retrieval system for causing a computer to retrieve data being stored in a database, said retrieval system comprising:

a database storing data as a vector digitized based on a keyword;

> At step 16, the disclosed system receives a user query from a user, consisting of a list of keywords or phrases. The disclosed system parses the electronic text included in the received user query at step 16. The parsing of the electronic text performed at step 16 may include, for example, recognizing acronyms, extracting word roots, and looking up those previously generated concept ID numbers corresponding to individual terms in the query. In step 17, in response to the user query received in step 16, the disclosed system generates a user query vector having as many elements as the number of rows in the term-spread matrix generated at step 9. (col 7, lines 27-38)
>
> Further as illustrated in FIG. 2, the indexing module 20 performs steps to reduce the original documents 27 and a query received from one of the clients 21 into symbolic form (i.e. a term-document matrix and a query vector, respectively). (Col. 8, lines 64-67)

a means for generating a residual vector from said data to compute a covariance matrix and an eigenvector of said covariance matrix, and for generating and storing a set of basic vectors from a set of said computed eigenvectors;

which has been amended thusly:

a means for generating a residual vector from said data, said residual vector corresponding to a vector in which an element corresponding to a contribution component in a direction of a basic vector calculated is subtracted from a previously obtained residual vector, the basic vector and the residual vector a newly generated residual vector lie in an orthogonal relationship, to compute a covariance matrix and an eigenvector of said covariance matrix, and for generating and storing a set of basic vectors from a set of ~~said~~ computed eigenvectors;

> Following creation of the query vector at step 17, at step 18 the disclosed system generates, in response to the user query vector, an error-covariance matrix. The error-covariance matrix generated at step 18 reflects an expected degree of uncertainty in the initial choice of terms by the user, and contained within the user query. (Col. 7, lines 39-45)

S.sub.ij is the covariance matrix of the errors n.sub.i in the input query vector, computed as S.sub.ij =Covar[n.sub.i,n.sub.j]=.delta..sub.ij n.sub.i.sup.2, if we assume that the errors n.sub.i on the elements of the input query are independent. By allowing for errors in the input query vector, which is based on the terms in the original query, the present system attaches a margin of uncertainty to the initial choice of terms input by the user. Since the user's initial term selection may not be optimal, the present system advantageously allows for a margin of error or a certain degree of flexibility in this regard.
(Column 14, lines 26-34)

LSI transforms the matrix D as D'=U.sub.k.LAMBDA..sub.k V.sub.k.sup.T where .LAMBDA.=diag(.lambda..sub.1, . . . ,.lambda..sub.k) and {.lambda..sub.i,i=1,k} are the first k ordered singular values of D, and the columns of U.sub.k and V.sub.k are the first k orthonormal eigenvectors associated with DD.sup.T and D.sup.T D respectively. From this we see that .omega.=(U.LAMBDA.).sub.k and A=V.sub.k.sup.T {A.sub.j, j=1,2, . . . ,n}.(Col. 12, lines 1-7)

FIG. 5 shows clustering for the document collection reflected by the table of FIG. 4, as obtained using an LSI approach, as in some existing systems. The dots in each of the graphs in FIG. 5 are plane projections of individual documents into "concept space", as determined by a choice of the first few eigenvectors. Documents which deal with similar topics cluster together in this space. (Col. 11, lines 20-25)

a means for reading out said data and at least one of said eigenvectors from a memory, and for computing a contribution of said eigenvector to said data, and for contracting or enlarging a residual vector to store; and

which has been amended thusly

a means for reading out said data and at least one of said eigenvectors from a ~~memory~~ database, and for computing a contribution vector of said eigenvector to said data, and for contracting or enlarging a residual vector to store;

FIG. 5 shows clustering for the document collection reflected by the table of FIG. 4, as obtained using an LSI approach, as in some existing systems. The dots in each of the graphs in FIG. 5 are plane projections of individual documents into "concept space", as determined by a choice of the first few eigenvectors. Documents which deal with similar topics cluster together in this space. (Col. 11, lines 20-25)

where w=A.sup.T.alpha. is the smallest l.sub.2 norm solution to the linear system Dw=q. Reducing the number of eigenvectors in the approximation

to the inverse of D has a regularizing effect on the solution vector w, since it reduces its norm. (Column 12, lines 15-18)

S.sub.ij is the covariance matrix of the errors n.sub.i in the input query vector, computed as S.sub.ij =Covar[n.sub.i,n.sub.j]=.delta..sub.ij n.sub.i.sup.2, if we assume that the errors n.sub.i on the elements of the input query are independent. By allowing for errors in the input query vector, which is based on the terms in the original query, the present system attaches a margin of uncertainty to the initial choice of terms input by the user. Since the user's initial term selection may not be optimal, the present system advantageously allows for a margin of error or a certain degree of flexibility in this regard. (Column 14, lines 26-34)

a means for selecting a keyword to be used for labeling according to a similarity between said stored basic vector and said data, and a weight on said similarity so as to store the keyword in a memory.

Which has been amended thusly:

a means for selecting a keyword to be used for labeling <u>clusters</u> according to a similarity between said stored basic vector and said data, and a weight on said similarity so as to store the keyword in a ~~memory~~ <u>database; means for classifying data into clusters of documents having the same or similar keywords and depending on a similarity between the stored basic vector and the data;</u>

FIG. 4 shows an example of a term-document matrix 40, and also illustrates some of the difficulties associated with existing systems. The term-document matrix 40 of FIG. 4 is shown, for purposes of illustration, loaded with word counts for 16 keyword terms (rows 42) in 15 documents (columns 44). The example of FIG. 4 illustrates testing of latent semantic retrieval. Topics present in document collection of FIG. 4 are "GEOGRAPHY" (documents b3, b4, b6 and b12), "THEATER" (b1, b5, b8, b9, b10, and b15), and "SHAKESPEARE" (b7 and b11). The keyword "Shakespeare" appears only in documents b7 and b11. The documents semantically related to the "THEATER" topic, however, may also be relevant to a search query which includes the single keyword "Shakespeare".

FIG. 5 shows clustering for the document collection reflected by the table of FIG. 4, as obtained using an LSI approach, as in some existing systems. The dots in each of the graphs in FIG. 5 are plane projections of individual documents into "concept space", as determined by a choice of the first few eigenvectors. Documents which deal with similar topics cluster together in this space. The key to successful semantic retrieval is to select a subspace where documents 54 which contain the keyword "Shakespeare" cluster as a subset of all documents 56 which deal with the topic of "THEATER". This is the case for the two projections shown by the graphs 50 and 52, but not for graphs 51 and 53. Graphs 51 and 53 in FIG. 5 are examples where the "SHAKESPEARE" documents 54 do

not appear as a subcluster of the "THEATER" documents 56. Graphs 50 and 52, on the other hand, are examples where the "SHAKESPEARE" documents 54 appear as a subcluster of the "THEATER" documents 56. It is difficult to predetermine which choice of projection axes x-y that will cause the desired effect of clustering the "SHAKESPEARE" documents as a subcluster of the "THEATER" documents. More specifically, it is difficult to predetermine how many eigenvectors--and which ones--one should use with LSI in order to achieve this result. FIG. 5 illustrates that there is no way of pre-determining the combination of axes which cause the "SHAKESPEARE" documents to appear as a subcluster of the "THEATER" documents. (Column 11, lines 4-46)

The graph 62 of FIG. 6 illustrates the comparison of the desired output query q (solid line 63) and the computed output query q' (undistinguishable from q) for the l.sub.2 -norm minimizing solution. The output q' is computed as a linear superposition of the first seven bases (also shown in FIG. 6), ordered by decreasing coefficients .vertline..alpha..sub.i.vertline.. Bases with positive .alpha..sub.i (basis 1 and basis 2) are shown with continuous lines. Bases with negative .alpha..sub.i (basis 3, basis 4, basis 5, basis 6, and basis 8) are shown with dotted lines. The positive bases contain primarily the input query keyword and contribute significantly to the query approximation. They also contain several other keywords (e.g. "theatre", "comedy") which are directly associated with the keyword "Shakespeare" across the document collection. These associated keywords must be subtracted in order for the approximation q' to match the desired output q. The negative bases accomplish this. The negative bases define partitions (or groups) of documents that contain many of the same keyword patterns found in the positive bases, this time never in direct association with the keyword "Shakespeare". Consequently, the negative bases span the space of the latent semantic documents. Latent semantic documents are documents that, while not containing any of the keywords in the user query, may contain a statistically significant number of keywords conceptually related to the keywords in the user query. (Column 15, lines 25-50)

Specifically, at step 14, the original term-document matrix created at step 6 and potentially weighted at step 8, rather than the term spread matrix computed at step 9, is cross-multiplied with the unsorted document weights generated at step 12 (note that the document weights must be unsorted in this step to match the original order of columns in the term-document matrix) to form a plurality of term weights, one for each term. These term weights reflect the degree of correlation of the terms in the lexical knowledge base to the terms in the user query. (Column 8, lines 23-32)

The following new clause has been added:

<u>and a means for outputing cluster data of a cluster to a graphical user interface system for displaying the cluster data.</u>

Summary

Marchisio's disclosure of negative bases to define partitions neither teaches nor suggests a contribution vector (defined in connection with an inner product calculated between a calculated vector and a document vector) to calculate a residual matrix to enhance document retrieval.

This is positively recited in the clause

> *"... a means for generating a residual vector from said data, said residual vector corresponding to a vector in which an element corresponding to a contribution component in a direction of a basic vector calculated is subtracted from a previously obtained residual vector, the basic vector and the residual vector a newly generated residual vector lie in an orthogonal relationship, to compute a covariance matrix and an eigenvector of said covariance matrix, and for generating and storing a set of basic vectors from a set of ~~said~~ computed eigenvectors; ... "*

but is neither taught nor suggested by Marchisio.

Conclusion

It is respectfully submitted that the Figures, the Specification, and all of the claims are proper under 35 USC §112 (second paragraph).

Based on the above discussion, it is respectfully submitted that the pending claims describe an invention that is statutory subject matter and is properly allowable to the Applicants.

If any issues remain unresolved despite the present amendment, the Examiner is requested to telephone Applicants' Attorney at the telephone number shown below to arrange for a telephonic interview before issuing another Office Action.

Applicants would like to take this opportunity to thank the Examiner for a thorough and competent examination and for courtesies extended to Applicants' Attorney.

Certificate of Mailing

I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being deposited with the United States Postal Service on the date shown below with sufficient postage as Certified Priority Mail (Return Receipt Requested), Certified Label, in an envelope addressed to the Commissioner for Patents, Alexandria Virginia, 22313

Date of deposit: October 5, 2006

Person mailing paper: Richard M. Goldman

Signature:

Respectfully Submitted

Richard M. Goldman, Reg. # 25,585
371 Elan Village Lane, Suite 208
San Jose, CA 95134
Voice: 408-324-0716
Fax: 408-324-0672
E-mail: goldmanptn@aol.com